Amendments to the Specification:

Please amend the two paragraphs beginning on line 7 of page 5 as follows:

~~Figures 2a-2e illustrate various embodiments of configuring a processing relationship structure that may be modeled after an FSO business organization structure;~~

Figure ~~2f~~2 is an example of one embodiment of a multilevel business processing relationship to be modeled in an FSO business system;

Please amend the portion of the specification beginning on line 23 of page 16 as follows:

~~Figures 2a-2f~~ Various embodiments of configuring a processing relationship structure that may be modeled after an FSO business organization structure

A Financial Service Organization (FSO) is a business organization that provides financial products and/or services to customers and/or client organizations. An FSO may include one or more organizational units. Examples of organizational units include, but are not limited to, an entity, a business unit, a subsidiary, a division, a functional unit, a headquarters, an operating unit, a profit center, a regional office, and a branch office.

~~Figure 2a illustrates an~~ In one example of an FSO business organization according to one embodiment. ~~For example,~~ the FSO business organization may be a global bank 2250. The FSO business units may be represented in a chart or a similar graphical form to illustrate the attributes of an FSO organization such as, but not limited to, the reporting relationship between various FSO entities, the reporting structure, the number of hierarchical levels between the highest level entity and the lowest level entity, and the number of direct reports for an FSO entity. Each FSO entity may be represented as a node or a block on an FSO organizational chart. For example, global bank ~~is~~ may be represented as node 2250, the business unit for Americas by node 2252, the business unit for Europe, Middle East and Africa by node 2254. Each node may have a parent node and one or more children nodes. For example, USA business unit 2256 ~~has~~ may have a parent node Americas 2252 and ~~has~~ may have two children nodes, region AUE 2260 and region AUW 2258. Each node may be identified uniquely with a node number and/or a name. The FSO organizational chart may include multiple levels 2266 in the hierarchical relationship. A node without a parent may be described as a root node or a level zero node. A root node may include the entire FSO organization. The global bank node 2250 may be described as a root node. The FSO organizational chart may be updated, in real-time, as new FSO entities are introduced or

2

removed by adding or deleting a node corresponding to the FSO entity. The FSO organizational chart may thus graphically represent the current, real-world state of the FSO organization.

Please amend the portion of the specification beginning on line 16 of page 18 as follows:

In one embodiment, the processing relationship structure 2276 may be represented graphically on a display screen 2270, as illustrated in Figure 2b. A user of an FSO may modify or edit the processing relationship structure 2276 by adding or deleting a node, e.g. the object associated with the node. In one embodiment, the node or object may be represented on a display screen 2270 as an icon or a symbol. In one embodiment, a group of objects, each represented as an icon, may be displayed as palette of objects 2274 on a display screen. In one embodiment, the user may use drag-and-drop techniques to add a new object selectable from a palette of objects 2274 to the processing relationship structure. For example, the FSO user may position a cursor 2268 on a node object 2274 and use a drag-and-drop method 2272 to place the selected object 2274 on the processing relationship structure. The FSO user may then configure the node, e.g., the object, by using and/or defining the properties and methods associated with that node.

In one embodiment, the processing relationship structure may be based on traditional programming and traditional database technology. Programming in the C language may be an example of traditional programming. Examples of traditional database technologies may include, but not limited to, hierarchical, proprietary, relational, flat file. Each node in the processing relationship structure may be represented, in one embodiment, by a table in a relational database. A node may be defined by the rows and columns associated with the table. For example, in one embodiment, a bank table may represent a node. The bank table may include attributes such as, but not limited to, a node identifier, a level number, a sequence number, a bank location identifier, an ATM location description, a customer account number, a type of loan. Access to the bank table may include identifying required keys such as, but not limited to, a transaction identifier, an account number, an FSO user identifier. In one embodiment, the processing relationship structure may be represented by text on a display screen 150, as illustrated in Figure 32c-d. The parent/child or a precedent/descendent relationship may be defined in one embodiment by defining a previous node identifier and a next node identifier. An FSO user may modify or edit the processing relationship structure by adding or deleting a row in a table associated with the node being edited. The columns 152-162 shown in Figures 2c-2e are further described with reference to Figure 3. The FSO user may add the root level node 2250 in Figure 2e. In one embodiment, the FSO user may add a first row to a global bank table. The user may configure the processing relationship structure by entering values for attributes such as, but not limited to, a node identifier, a level number, a sequence number. In Figure 2d, tThe FSO user may insert a row to add node 2252 for Americas. The user may configure the new node by

3

entering values for attributes such as, but not limited to, a node identifier, a level number, a sequence number. ~~In Figure 2e, t~~The FSO user may insert a row to add node 2254 for Europe, Middle East and Africa. The process may be repeated for all of the remaining nodes included in the global bank business organization chart ~~in Figure 2~~a. The FSO user may perform a modification to the processing relationship structure, e.g., may reconfigure based on changes in the real world.

Please amend the portion of the specification beginning on line 18 of page 20 as follows:

The processing relationship structure may be used by FSO application software programs to process FSO transactions. Examples of application software which may utilize the processing relationship structure, may include, but are not limited to, a report generation program, a credit card transaction processing program, a billing program, a monthly account reconciliation summary program. In one embodiment, changes made to the node associated objects and/or tables may have little or no effect on the application software program source code. For example, ~~in Figure 2~~a the global bank may reorganize its visa account business unit 2262 such that the visa unit now falls under region AUW instead of region AUE. This change may have little or no impact on the report generation program source code for the visa account business unit 2262 since all the objects and/or tables associated with the visa account node, i.e., the owner of the data, may be automatically updated when the FSO user makes changes to the processing relationship structure. The application programs may reference the current properties and/or attributes of the node objects and/or tables to process FSO transactions.

Figure 2f through Figure 9 further illustrate various embodiments of configuring a processing relationship structure by starting with a representative FSO organization structure in Figure 2f and ending with a corresponding processing relationship structure in Figure 9. Figures 10a-10d include various flow charts illustrating one embodiment of a method of configuring processing relationships for use in an FSO application software program, such as a report program.

Figure 2f - An example of one embodiment of a multilevel business processing relationship to be modeled in an FSO business system

Figure 2f graphically illustrates one example of a multilevel business processing relationship that may be modeled in an FSO business system using a processing relationships configuration program according to one embodiment. An FSO user or any other person or persons familiar with the FSO organization may create a graphical diagram similar to Figure 2f to reflect the FSO business organization.

4

Please amend the portion of the specification beginning on line 1 of page 23 as follows:

By using a processing relationships configuration program and its associated display screens, as described in Figures 3-9, the FSO user may configure the processing relationship structure. At the end of the configuration process, Figure 9 may describe a processing relationship structure, which may be equivalent to the multilevel business processing relationship illustrated in Figure 2f.